

Uvod in splošno o omrežjih

Zgodovina računalniških komunikacij

Osnovni definiciji

Računalniško omrežje

Računalniško omrežje tvorijo dva ali več računalnikov, ki so med seboj povezani, a neodvisni.

Porazdeljen sistema

Porazdeljen sistem je decentraliziran sistem, kjer je s stališča uporabe obstoj množice povezanih, a sicer avtonomnih sistemov za uporabnika transparenten.

Primer: Števec porabe električne energije. Števci so med seboj povezani, vsak zase pa je avtonomen (števec bo še vedno štel porabo tudi, če ni priključen na omrežje).

Zgodovina računalniških komunikacij

Neke karakteristike današnjih računalniških komunikacij so se že kazale v telegrafskih mrežah v 19. stoletju, vendar o začetku računalniških komunikacij lahko govorimo šele s pričetkom terminalskih omrežij.

Pri terminalskih omrežjih je šlo za nek velik centralni računalnik do katerega se je dostopalo preko terminalov, ki so služili samo kot tanek vmesnik med uporabnikom in centralnim računalnikom, torej kot neka omrežena vhodno izhodna naprava.

S pojavom osebnih računalnikov so se pa začele razvijati mreže kot jih poznamo danes. Thomas J. Watson je rekel, da bomo v prihodnosti rabili kvečjemu par računalnikov na svetu, ki bodo računali vse. V neki meri se je to izkazalo za resnično, saj so več ali manj vsi osebni računalniki povezani preko interneta v en porazdeljen sistem. Primeri izkoriščanja tega sistema za računsko moč so recim projekti SETI@home in folding@home.

Danes se vse več pojavljajo tudi naprave z mikroprocesorji, ki so manjše in bolj prenosne ter komunicirajo ena z drugo ter z računalniki. Edina omejitev za to je velikost tranzistorjev in posledično teh naprav. Vse več naprav komunicira med seboj.

Na kratko

- osrednji računalnik + terminalska omrežja
- povezovanje terminalskih omrežij
- osebni računalniki in na njih temelječa omrežja (1980, mikronska tehnologija)
- povezovanje osebnih računalnikov v omrežja (izmenjevanje podatkov, združevanje dela)
- naprave z mikroprocesorji (izboljšanje tehnologije: nanometerska tehnologija)

Rast omrežij

- Omrežja ponavadi rastejo iz **lokalnih** v **globalna**. Lokalna omrežja se združujejo v krajevna omrežja, ki se potem združujejo in rastejo v globalna.
- Ker ima vse več različnih naprav možnost komuniciranja, se **homogena** omrežja spreminjajo v **heterogena**. Omrežja med katerimi so včasih bili samo računalniki sedaj vsebujejo mobilne telefone in vrsto različnih naprav, ki preko določenih protokolov lahko medsebojno komunicirajo.

Taksonomija - klasifikacija omrežij

Delitev glede na nizkonivojsko aparaturno opremo

Omrežja lahko ločimo glede na opremo, ki jo uporabljajo. Poznamo omrežja z **razpršenim oddajanjem** (ang. *broadcasting*). Primer bi bil lahko Eduroam omrežje na FRI ali Ethernet protokol (na hitro: računalniki povezani s topologijo vodila). Druga vrsta omrežij pa uporablja način povezave **točka-točka** (ang. *point to point*). Primer te vrste bi lahko bila topologija mesh ter obroč.

Značilnosti razpršenega oddajanja

Pri razpršenim oddajanju si vsi odjemalci delijo isti komunikacijski kanal.

Zaradi tega poteka vsa komunikacija na tem kanalu v **paketih**, saj lahko v času zatišja vsjoči nekdo drug ter prevzame vlogo **sporočevalca**. Dodatna posledica deljenja skupnega kanala je pa tudi ta, da **vsi** slišijo **vse**.

Izjema tega je **selektivno oddajanje** (ang. *multicasting*) pri kateremu določimo **manjšo** skupino prejemnikov. To storimo lahko **fizično** (uvedemo dodatno frekvenco) ali **logično** (dodamo podatek, ki pove za koga je namenjen dotični paket).

Značilnosti povezav Točka-točka

Pri temu načinu sami odjemalci **sodelujejo** pri prenašanju informacij, saj so med seboj povezani. Zaradi tega ni omejitev pri načinu pošiljanja, saj lahko en računalnik neprestano govori svojemu sosedu, seveda pri temu moramo predpostaviti da sta povezana z dvema povezavama (da dobimo **dvosmerno** komunikacijo).

Za uspešno komunikacijo med dvema računalniko ni potrebno, da sta sosedata. Seveda pri temu nastane problem **posredovanja sporočil**, saj obstaja **več možnih** poti. Ta problem se razreši z **usmerjevanjem** (ang. *routing*).

Delitev glede na topologijo

Eden od pogojev za uspešno komunikacijo med računalniki je ta, da jih je potrebno povezati, v neko **mrežo**. To lahko naredimo na več načinov, oziroma z različnimi **topologijami**:

- Obroč (ring)
- Zvezda (star)
- Vodilo (bus)
- Drevo (tree)
- Mrežna (mesh)

Obroč

Vsi računalniki so povezani med seboj. Velika slabost je, da v primeru izpada enega se prekine krog. Slabost se popravi na ta način, da se podatke(pakete) pošlje v drugi smeri kroga kot prej(pri izpadu). S številom dodajanja računalnikov hitrost prenosa pada.

Zvezda

V sredini "zvezde" je ponavadi stikalo (switch) lahko pa je tudi hub ali računalnik. Vsi računalniki so povezani s sredino (centralno vozlišče), kar omogoča hitro komuniciranje dveh računalnikov. Komuniciranje dveh računalnikov se naredi v maksimalno dveh korakih. Prvi korak je do centralnega vozlišča in drugi do ciljnega računalnika. Velika slabost je izpad centralnega vozlišča.

Po izpadu ne more nobeden od povezanih računalnikov komunicirati med seboj.

Vodilo

Pri tej topologiji so vsi računalniki povezani med seboj z enim kablom (ponavadi je to koaksialni) in pri njej lahko komunicirata le 2 računalnika hkrati, ostali pa morajo čakati, da končata "pogovor". Slabost te topologije je, da ob izpadu enega računalnika izpade celotno omrežje.

Drevo

Drevo je topologija, ki razdeli omrežje na manjša podomrežja, ta na še manjša itd. Slabost te topologije je, če izpade en člen, izpade iz glavnega omrežja tudi celotno podomrežje, ki je povezano z odpadnim računalnikom.

Delitev glede na deljenje komunikacijskega kanala

Več računalnikov si lahko razdelijo en kanal s pomočjo **frekvenčnega** ali **časovnega** multipleksiranja. Pri frekvenčnem multipleksiranju se računalniki razdelijo na več različnih frekvenc, dober primer bi bil brezžično omrežje IEEE 802.11.

Časovno multipleksiranje pa pomeni, da vsak odjemalec uporablja medij ob določenem času.

Deljenje medija je lahko **statično** ali **dinamično**.

Delitev glede na fizični obseg

Omrežja lahko tudi razdelimo glede na njihovo razsežnost oziroma **obseg**. Poznamo sledeča omrežja:

- **lokalna omrežja** (ang. *local area networks*), so ponavadi v domovih in obsegajo od nekaj metrov do 1 km;
- **mestna omrežja** (ang. *metropolitan area networks*), so na območju večjih mest in obsegajo okoli nekaj 10 km. Primer bi lahko bila kabelska televizija/internet (satelitska povezava na enem mestu, ki je potem razdeljena med večjim številom hiš);
- **širokopodročna omrežja** (ang. *wide area networks*), so med državami in kontinentni in obsegajo nekje med 100 do 1000 km;
- **globalna omrežja** (ang. *global area networks*).

Medmrežje

Medmrežje(Internet) je računalniško omrežje, ki povezuje več različnih omrežij skupaj. Pri tem povezavo omogoča takoimenovan gateway oziroma pretvornik. Vsem poznam primer je recimo povezava iz našega lokalnega omrežja (LAN) v Internet oziroma drugo (WAN). Pretvornik(gateway) se obnaša kot vrata v neko drugo omrežje.

Nastanek Interneta

Začne se v ZDA konec šestdesetih let (1969). V osemdesetih letih je agencija za napredne raziskave pri ameriškem obrambnem ministrstvu DARPA (Defence Advanced Research Projects Agency) razvila protokol TCP/IP (Transmission Control Protocol/Internet protocol). Glavni namen je bil omogočiti komunikacijo med različnimi mesti, državami v primeru jedrskega napada. Kasneje se je protokol razširil tudi v akademsko in raziskovalno omrežje, kmalu pa je postal uporaben za vse komercialne uporabnike.

Fizična plast

1. Uvod

Cilji:

- prenašanje bitov preko komunikacijskega kanala, skladno z omejitvami medija
- prenašanje bitov s čim manj napakami

Za prenos podatkov uporabljamo neke fizikalne lastnosti, ki predstavljajo 0/1 v našem bitu (sprememba električne napetosti, svetlobe , ...). Pomembno je, da je preprosto postaviti stanje 0 oz. stanje 1, ter da je preprosto razlikovati med obema stanjema. Katero fizikalno lastnost uporabljamo, je odvisno od medija, po katerem potuje signal. Za modeliranje signalov uporabljamo matematična orodja.

1. Analiza signalov - Fourierova vrsta

- Periodično funkcijo s periodo t lahko predstavimo s sinusi in kosinusi ($f=1/t$):

$$g(t) = \frac{c}{2} + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$

- Posamezne koeficiente (amplitude harmonikov) določimo z enačbami(uporabimo končno število členov (npr. 8)):

$$c = \frac{2}{T} \int_0^T g(t) dt$$

$$a_n = \frac{2}{T} \int_0^T g(t) \sin(2\pi nft) dt$$

$$b_n = \frac{2}{T} \int_0^T g(t) \cos(2\pi nft) dt$$

- Vsak medij različno dobro prenaša različne frekvence (popačenje signala);
- Motnje nastajajo zaradi fizikalnih lastnosti kanala);

Torej :

- po liniji pride signal analogno,
- mrežna kartica (ali kaka druga naprava) to vzorči, naredi inverzno Fourierovo transformacijo in dobi ven, kar smo na drugi strani poslali noter.

2. Komunikacijski kanal

- Prvi komunikacijski kanal - zrak;
- Kateri medij uporabiti:
 - o cena napram hitrosti
 - o vprašanje trajnosti
 - o vprašanje varnosti
- Prepustnost komunikacijskega kanala:
 - o število sprememb vrednosti signala v eni sekundi imenujemo baud (hitrost v baudih ni nujno enaka hitrosti v bitih na sekundo (odvisno koliko bitov lahko damo v en baud));
 - o če znamo opisati komunikacijski kanal na način, da dobimo iz tega zgornjo mejo števila prenesenih signalov v časovni enoti, potem lahko iz te zgornje meje ugotovimo, koliko baudov lahko pošljemo skozi komunikacijski kanal, iz tega pa število bitov na sekundo tako, da pogledamo koliko bitov je v enem signalu;
 - o kapaciteta kanala nam pove, koliko harmonikov gre čez;
 - o Nyquistov teorem:
 - $f_{vz} > 2H$
 - Hitrost vzorčenja mora biti dvakrat tako hitra, kot je hitrost fizikalnega fenomena (H predstavlja frekvenčno širino);
 - Če hočemo izmeriti H harmonikov, mora bit frekvenca vzorčenja vsaj 2H (glej OTI).
 - o Shannonovo pravilo:
 - govori o tem, kakšno mora biti razmerje S/N, da je stvar še zaznavna;

- kako visoka mora biti amplituda, da “preglasi” šum -> višja amplituda, več energije in tudi večje popačenje (večji čas, da pridemo od 1 do 0).

3. Mediji

- Magnetni medij:
 - o eden prvih
 - o exabyte trak (7GB)
- Parica:
 - o izolirana in prepletena bakrena žica,
 - o brez ojačevanja je uporabna na razdaljah nekaj km,
 - o cenovno ugodne, kategorija 3, 5, 7 (UTP),
 - o UTP – unshielded twisted pair. Kategorija 3 in kategorija 5 - pove koliko zasukljajev je na neko enoto. Včasih smo s temi kabli dosegali 100 Mbit/s, kasneje 1 Gbit/s, sedaj govorimo že o 10Gbit/s UTP kablil;
- Koaksialni kabel:
 - o Sestavljen je iz štirih delov: jedro, po katerem se prenašajo podatki, to pa je obdano z izolacijsko plastjo;
 - o Da je kabel zavarovan pred elektromagnetnimi valovanji iz okolja, sta notranji dve plasti oviti z zunanjim vodnikom v obliki pletenice, vse skupaj pa je še enkrat obdano z zunanjo izolacijo;
 - o Čeprav ima tehnične lastnosti veliko boljše od parice: hitrosti dosegajo od 1 do 2 Gb/s na km (kar je odvisno od njihove impedance) in odpornost na EMV, pa so zaradi njihove zgradbe trdi in neupogljivi, hkrati pa je draga tudi proizvodnja;
- Optično vlakno:
 - o Prednost: dosegljiva pasovna širina do 50 Tb/s (na kratkih razdaljah), in to z zelo majhno verjetnostjo napake: $10^{-5}/s$;
 - o Uporabljajo se za zelo dolge razdalje;
 - o Slabosti: Po strukturi je tudi optično vlakno koaksialno, zato ima tudi ta medij slabost v upogljivosti, poleg tega pa je tudi krhko + ozko grlo pretvorba v el. signal in preklapanje.
- Brezžične komunikacije:
 - o spekter med 10^4 do 10^{11} Hz,
 - o osnova za mobilne komunikacije,
 - o nizke frekvence prodirajo v stavbe, skozi ovire, moč pada s tretjo potenco oddaljenosti,
 - o visoke frekvence se širijo skoraj v ravni liniji, se odbijajo od ovir in se absorbirajo v dežju,
 - o najboljše pa je območje nad 100 MHz, kjer je razmerje S/N (signal/šum)največje;
 - o Satelitske komunikacije:
 - 2 vrsti satelitov: višje ležeči oziroma stacionarni. Ti imajo velike zakasnitve (latence) – okoli 270 ms, kar je slabost, vendar bi za pokritje površine zemlje rabili samo 3. Druga vrsta so nižje ležeči, ki imajo zakasnitev le od 1 do 7 ms, vendar jih za enak izkoristek potrebujemo okoli 50.
- Telefonski sistem.

Telefonski sistem:

- o je eden najbolj razširjenih komunikacijskih sistemov
- o Se sestoji iz končnih central, tranzitnih central in lokalne zanke
- o Ta omrežja so namenjena prenosu govora, zato problem kakovosti prenosa podatkov
- o Prve besede izrečene po telefonu: "Mr. Watson -- come here -- I want to see you."
- o Tehnologije:
 - Analogni signal je težko osveževati in njegova regeneracija vnaša napake (šum), zato se pojavlja le v lokalnih zankah.
 - Digitalni signal je enostavno osveževati in ni izgube informacij, oprema je cenejša - vse kar moramo, je pravilno prepoznati 1 in 0.
- o Tipične ključne komponente:
 - lokalna zanka (parica, do nedavno samo analogni signal),
 - spojni vodi (optika, mikrovalovi, dig. Signal),
 - telefonske centrale.

Lokalna zanka:

- o Problemi, ki se pojavijo pri prenosu signala prihaja do:
 - popačenja zaradi dušenja
 - popačenja zaradi zakasnitev in filtriranja harmonikov;
 - popačenja zaradi šuma (termičnega) in motenj.

Modem:

- o Modemi so naprave, ki iz digitalnega signala spremeni v analogni signal tako, da zna dekodirnik na drugi strani signal spet dekodirati
- o Digitalni signali imajo širok spekter, zato uporabljamo t.i. sinusni nosilec (1-2 kHz)
- o Modeme priključujemo preko serijskih vrat (RS 232) ali na vodilo (npr. PCI, ISA)
- o Nosilec moduliramo tako, da mu:
 - spreminjamo amplitudo in / ali
 - frekvenco in / ali

- fazo.

Modulacije:

- o Poznamo 4 vrste modulacij:
 - Amplitudna (spreminjanje jakosti signala, občutljiva na motnje in enostavna izvedba)
 - Frekvenčna (spreminjanje frekvence signala, hitrost omejena z širina kanala, še bolj občutljiva na motnje)
 - Fazna (uvedba faznih zamikov in nujna sinhronizacija)
 - Kombinirane modulacije (Ponazoritev z diagrami, ki podajajo kombinacije amplitud in faz – konstalacijski vzorci.
- o Poznamo QPSK, QAM-16, QAM-64, V.32, V32

DSL:

- o Pas v katerem komunicira je tako imenovani 256 4-kHz kanalov, neodvisnih en od drugega
- o Kanal je treba izkoristiti, zato je več kanalov za downstream, kakor za upstream, ker uporabnik običajno porabi več prenosa k uporabniku, kot od uporabnika stran, si pa vsak uporabnik to po svoji želji nastavi, takrat govorimo o ADSL, kjer črka A stoji za asimetričnost.

Multipleksiranje:

- o Poznamo 4 možnosti:
 - statistično (polnimo kanal samo z tistimi signali, ki so aktivni)
 - časovno (nižje pasovne povezave poveže v višje pasovne $T_2=4T_1$)
 - frekvenčno (združi signale na različnih frekvencah)
 - valovno (več valov gre na eno vlakno)

Vrste switching-a (način povezave dveh naprav, ki hočeta med seboj komunicirati) :

- o Circuit (povezavna)
- o Packet (paketna)
- o Message (sporočilna)

Enota	Povezavna	Paketna
Klicne nastavitve	Potrebne	Nepotrebne
Namenske fizične poti	Da	Ne
Vsak paket gre po enaki poti	Da	Ne
Paketi pridejo v poslanem zaporedju	Da	Ne
Je za stikalno trčenje usoden	Da	Ne
Pasovna širina na voljo	Fiksna	Dinamična
Kdaj lahko pride do zastojev	Ob nastavitvi časa	Na vsak paket
Potencialno zahtevana pasovna širina	Da	Ne
Shrani in posreduje naprej	Ne	Da
Transparentnost	Da	Ne
Zaračunavanje	Na minuto	Na paket

Linajska plast

Linajski sloj nudi fizičnemu sloju *zanesljiv komunikacijski kanal*. Bite, ki se prenašajo razstavi na **bloke** oz. **okvirje**, ti pa se nato prenašajo po fizičnem sloju

Namen linajskega sloja je torej rešiti naslednje **težave**:

- pojav napak
- komunikacija med različno zmogljivimi računalniki
- različne zakasnitve pri prenosu

Tri osnovne vrste storitev linajskega sloja:

- **Nepovezavna brez potrjevanja**
 - o primerna za zanesljive kanale, kjer se pojavlja malo napak
 - o ustrezna za storitve, ki potrebujejo prenos v realnem času, npr. *govor*
- **Nepovezavna s potrjevanjem**
 - o vsak okvir je individualno potrjen
 - o primerna je za manj zanesljive medije, npr. *brezžične*
- **Povezavna s potrjevanjem**
 - o med osebkoma se vzpostavi povezava, nato pa sledijo podatki in njihove potrditve
 - o vsak okvir je prejet le enkrat in v določenem zaporedju

Faze povezavne storitve z uporabo potrjevanj

Pri povezavnih storitvah so vedno prisotne tri faze. Nastopajo lahko časovno ločeno ali pa hkrati. Faze so:

1. Vzpostavitev povezave

- o zahteva inicializacijo spremenljivk na obeh straneh (npr. štetje okvirjev, inicializacijo časovnikov, zasedanje pomnilnika)
- 2. **Prenos podatkov** z oštevilčevanjem in potrjevanjem ter ponovnim pošiljanjem
- 3. **Zaključek povezave**
 - o pomeni sprostitve vseh resursov (spremenljivk, medpomnilnika, procesorskega časa)

Uokvirjanje

Linijski sloj mora obvladovati neobdelane nize bitov, zato jih preoblikuje v smiselne enote – **okvire**(angl. Frame). Okvir je osnovna podatkovna enota linijskega sloja. Vsak okvir ima svoj:

- **Začetek** (Glava)
- **Vsebino** (Podatki)
- **Konec** (Rep)

Ob tvorjenju okvira se v *glavo okvira* vstavi *število znakov* v okviru. Problem se lahko pojavi pri napačnem prenosu glave, saj se v glavi pojavi napačno število znakov, kar privede do izgube sinhronizacije. Probleme predstavljajo tudi časovne razmejitve, saj lahko omrežje spremeni njihovo trajanje.

Tvorjenje okvirjev

Poznamo mnogo načinov tvorjenja okvirjev. Najbolj pogosto uporabljen je IBM-ov binarni sinhroni nadzor. Okvirje tvorimo z posebnimi ASCII znaki. Začetek paketa je označen z znakom DLE(Data Link Escape), ki mu sledi znak STX(Start of Text). Nato sledijo podatki, na koncu pa zopet damo znak DLE ter znak ETX(End of Text). Problem nastane če imamo tudi v podatkih znak DLE. Potem moramo pred ta znak vriniti še en znak DLE, tako, da lahko vedno ločimo podatke od glave oziroma repa.

V praksi se pogosto uporabljajo tudi protokoli za tvorjenje bitno orientiranih okvirjev. Pri tvorjenju bitno orientiranih okvirjev moramo označiti začetek in konec okvira. Za to se uporablja bitna sekvenca 01111110. Tako sprejemnik ugotovi, kdaj se prenos okvira začne in kdaj se prenos okvira konca. Seveda se lahko tudi to pojavi med podatki bitni vzorec, ki pomeni konec okvira. Težava se reši na način, da oddajnik pri pošiljanju podatkov po vsaki peti zaporedni enici vrine dodatno ničlo. Če sprejemnik naleti na sekvenco petih enic in ničle (111110), zadnjo ničlo izloči iz podatkov. Zaporedje šestih enic pa predstavlja začetek ali konec okvira. Tako se zagotovi transparentnost podatkov.

Zapiske pripravili (Za predavanja dne 2.3.2009 13h-14h): Matevž Bizjak, Martin Artnik, Matjaž Čuk, Jan Berdajs

Hammingova koda

Vzamemo 7 bitno kodno besedo, ki je sestavljena iz podatkovnih in nadzornih bitov.

Število podatkovnih bitov je enako m , število nadzornih je enako r , skupno število bitov pa je n . (V našem primeru je $n=7$, $m=4$ in $r=3$.)

Hammingovo matriko H množimo s kodirnim vektorjem v , rešitev enačbe $H*v=0$ je kodna beseda. Od tod dobimo sistem linearnih enačb, s katerimi izračunamo x , y in z ter jih vstavimo v vektor v . Dobljeno kodno besedo pošljemo v komunikacijski kanal.

Če rezultat pri množenju vektorja s Hammingovo matriko ni enak 0, pomeni, da je prišlo do napake. Končna vrednost nam pove, kateri bit se je dejansko spremenil.

Prenašamo n bitov, nadzorni biti so na mestih 2^i , veljati mora $2^r \leq n$.

$$m = n - \log_2 n$$

Učinkovitost Hammingove kode pa je enaka

$$(n - \log n)/n$$

Kode za obvladovanje napak

Način s popravljanjem napak je nesmiseln, če ni veliko napak. Pravi rezultat dosežemo s ponovnim prenosom.

CRC – ciklične redundančne kode (Cyclic Redundancy Code)

Bite v nizu predstavimo s polinomom. 110001 $\Rightarrow 1*x^5 + 1*x^4 + 0*x^3 + 0*x^2 + 0*x + 1$

Polinom zazna napako, ki pa jo lahko zgreši, če se napaka pojavi v vzorcu, zato ga moramo pravilno izbrati. Hočemo ujeti tiste napake, ki se najpogosteje pojavljajo.

Napaka se pojavi v kanalu, ki je fizično realiziran. Vsak kanal ima svoje tipične napake, ki jih lahko predvidimo. Zato si izberemo takšen polinom $G(x)$, da deljenje pri tipičnih napakah ne bo dalo ostanka 0.

Standardizirani generatorji so CRC-12, CRC-CCITT.

Krmiljenje pretoka

Razmerje med pošiljateljem in naslovnikom se imenuje nadzorovanje pretoka. Procesi oddajanja in sprejemanja so nepovezani.

Preprost enosmerni protokol brez omejitev:

- procesni čas je zanemarljiv,
- na fizičnem nivoju je idealen komunikacijski kanal ne prihaja do napake
- ni zaporednih števil ali potrditev,

- podatki tečejo le v eni smeri,
- neskončni medpomnilniki,
- mrežna sloja vselej zmožna sodelovati.

Potek protokola:

- Oddajnik brez prestanka pošilja podatke.
- Sprejemnik le čaka na dogodek, ki je vedno le sprejem neoporečnega okvirja in preda vsebino mrežnemu nivoju.

Sčasoma bomo dodajali omejitve.

Ne sprejemnik ne oddajnik ne moreta neskončno hitro pošiljati oziroma sprejemati podatkov (nista enako hitra). To lahko rešimo na dva načina:

- Upočasnjeno delovanje oddajanja.
- Prejemnik pošlje potrditveni paket oddajniku, ko je sposoben sprejemati podatke. Uvedemo half-duplex kanal, po katerem v drugo smer prihajajo samo potrditve.

Enosmerni “stoj in čakaj” protokol - Simplex Stop-and-Wait Protocol (protocol 2)

Ker ima prejemnik omejeno količino medpomnilnika in hitrost procesiranja, mora protokol poskrbeti, da pošiljatelj ne pošilja okvirov hitreje kot jih lahko prejemnik sprejme. To dosežemo tako, da se uvede ustavljanje in čakanje na potrditev sprejema okvira.

Ko prejemnik prejme poslani okvir, pošlje pošiljatelju potrditev, da je prejel okvir. To imenujemo Half-Duplex način. V tem času je kanal neizkoriščen.

Problemi:

- V primeru, da se poslani okvir izgubi (zaradi šuma), ga prejemnik ne prejme in ne pošlje potrdila, pošiljatelj pa čaka na potrditev, in ne pošlje novega okvira
- Okvir pride, potrditev se izgubi, pošiljatelj ne pošlje novega okvira

Enostavni protokol za šumni kanal - A Simplex Protocol for a Noisy Channel (protocol 3 - PAR)

Tu začnemo upoštevati šum, ki povzroči da se okvir med prenosom izgubi.

Rešitev:

- Dodamo časovnik, ko pošljemo paket, čakamo določen čas na potrditev, če ta v tem času ne pride, ponovimo prenos. Čas določimo glede na razdaljo med prejemnikom in oddajnikom.
- Prejemnik mora vedeti kateri okvir pričakuje, da loči med novim in ponovno poslanim okvirom. Ta spremenljivka se poveča, ko prejemnik dobi nov okvir.
- Pošiljatelj mora vedeti kateri okvir bo poslal kot naslednjega. V primeru da ne dobi potrditve za prejšnji okvir, tega ponovno pošlje. Ta spremenljivka se poveča, ko dobi potrditev za poslani okvir.

Problemi:

- Ko se okvir pošlje in čakamo na potrditev, je kanal neizkoriščen.
- Pri zanesljivih medijih je veliko število potrditev nepotrebnih.

Protokol drsečega okna

Zaradi želje po večji izkoriščenosti kanala uvedemo Full-Duplex, kjer se podatki prenašajo v obe smeri naenkrat. Pri tem se potrditve vključijo v posebna (ack) polja povratnih podatkovnih okvirov. Možnosti začasne zakasnjene potrditve, ki se vključi v prvi razpoložljivi podatkovni okvir, pravimo piggybacking.

Okviri so oštevilčeni s številkami od 0 do $2^n - 1$. Pošiljatelj odpre okno in pošlje okvir prejemniku. Prejemnik odpre okno, ko prejme okvir pošlje potrditev pošiljatelju in svoje okno zapre. Ko pošiljatelj prejme potrditev, svoje okno zapre. Pošiljateljevo okno vsebuje številke okvirov, ki so bili poslani, a še niso potrjeni. Prejemnikovo okno vsebuje številke okvirov, ki jih ta lahko sprejme v nekem trenutku. Okna so lahko fiksna ali dinamična.

Drseče okno velikosti 1 (protocol 4)

Oseba A pošlje osebi B poleg podatkov, ki jih pošilja še zaporedno številko okvira in številko okvira, ki ga potrjuje. Oseba B ta okvir prejme in pošlje osebi A podatke, vključno z zaporedno številko okvira, ki ga pošilja in zaporedno številko okvira, ki ga potrjuje.

Pri drugi možnosti začneta obe osebi pošiljati podatke naenkrat. V tem primeru ni dvojniku, a je veliko ponovitev. Tu nastopi problem na začetku, saj zgloda kot da so bili podatki izgubljeni, a niso bili.

Čas prenosa

$$\text{Izkoristek} = I/(I+bR)$$

I...čas prenosa paketa

b...pasovna širina

R...krožni čas

Cevovodenje

V praksi zakasnitve niso zanemarljive, npr. pri satelitih, kjer sta tako pasovna širina kot zakasnitveni čas velika. Problem je v čakanju na potrditev.

Rešitev je v tem, da pošiljatelj pošlje več okvirov naenkrat (pri satelitih 26), preden čaka na potrditev (to je 26. okvir, prejšnjih 25 je podatkovnih). Sklepamo, da je prejemnik vse poslani okvire dobil, in če pošiljatelj prejme potrditev, je vse v redu. Če ne prejme potrditve, ponovi prenos.

Protokol Go Back N »vrni se n okvirov« (protocol 5)

Problem nastane, če se med prenosom kateri izmed okvirov poškoduje ali izgubi. Če po preteku časovnika pošiljatelj ne prejme potrdila, da je bil nek okvir sprejet, se sprejmejo le okviri do poškodovanega, vsi ostali se morajo poslati ponovno.

Ponovno selektivno pošiljanje – PSP (selective repeat protocol, protocol 6)

Pri tem protokolu se ponovno pošlje samo okvir, pri katerem je prišlo do napake. Ostali se sprejmejo in shranijo v medpomnilnik (buffer), na koncu pa se vsi okviri razvrstijo v pravilni vrstni red. Za vsak nepotrjen okvir potrebujemo svoj časovnik. Medpomnilnik mora biti dvakrat večji od velikosti okvira, da se prepreči prekrivanje.

SLIP (Serial Line IP)

Določa ga RFC 1055.

Je zelo preprost, ne obvladuje napak, od optimizacij uporablja le stiskanje glave, ni overjanja.

Delovanje: IP paketom dodamo končno zastavico 0xC0. Če se ta znak pojavi v paketu se nadomesti z zaporedjem 0xDB, 0xDC. Če se v paketu pojavi 0xDB, se tudi ta polni.

PPP (Point2Point? Protocol)

Določa ga RFC 1661.

Obvladuje napake, podpira več protokolov, omogoča overjanje in dinamično dodeljevanje IP naslovov. Delovanje: pakete popolno uokviri z začetnim in končnim nizom (ta vsebuje tudi kodo za obvladovanje napak).

Vsebuje LCP (Link Control Protocol), ki povezavo postavi, preveri, izvede pogajanja in jo razdre.

NCP (Network Control Protocol) – možnost pogajanja o opcijah mrežnega sloja za različne protokole.

Struktura okvira: zastavica, naslov, kontrola, protokol, up. podatki

Lahko ga predstavimo s končnim avtomatom kot fazni diagram za vzpostavitev povezave med modemi in usmerjevalniki.

[Linjska plast in dostop do medija](#)

Wavelength Division Multiple Access Protocols

Kanal lahko s pomočjo frekvenčnega, časovnega ali obeh multipleksiranj razdelimo na več podkanalov. To omogoča istočasno večkratno komuniciranje s pomočjo različnih valovnih dolžin. Eden takšnih protokolov je WDMA, ki so ga definirali leta 1992 in se uporablja pri optičnih povezavah.

Za optično omrežje potrebujemo dva kabla iz vsake postaje. Eden služi oddajanju, drugi prejetanju. Skupaj so kabli zlit v cilindar tako, da vsak kabel za sprejem vidi vse kable za oddajanje.

V protokolu WDMA ima vsaka postaja dva kanala (to dosežemo z različnimi valovnimi dolžinami):

- Kanal za podatke
- Kontrolni kanal

Vsak kanal je razdeljen v neko število časovnih rež, na koncu pa sledi reža s podatki o prostosti ostalih rež (statusna reža). Te reže se ves čas ciklično ponavljajo. Ves sistem je sinhroniziran z globalno uro.

Protokol podpira tri vrste prometa:

- Konstanten pretok podatkov s povezavno komunikacijo (mekompresiran video)
- Raznolik pretok podatkov s povezavno komunikacijo (prenos podatkov)
- (prenašanje UDP paketov)

Ideja povezavnega komuniciranja je ta, da naprava, ki želi komunicirati to sporočilo pusti v prosti reži kontrolnega kanala naprave s katero želi komunicirati. Vsaka naprava potrebuje dva oddajnika in dva sprejemnika:

- Oddajnik fiksne valovne dolžine za pošiljanje podatkov
- Oddajnik nastavljive valovne dolžine za pošiljanje na kontrolne kanale drugih postaj
- Sprejemnik fiksne valovne dolžine za poslušanje lastnega kontrolnega kanala
- Sprejemnik nastavljive valovne dolžine za prejemanje podatkov od ostalih postaj

Potek povezave – A želi komunicirati z B:

- A posluša podatkovni kanal od B, da izve katere reže so še proste
- A nato pošlje B-ju »connection request« za želeno prosto režo
- B prejme zahtevo po kontrolnem kanalu in A-ju dodeli prosto režo
- V statusni reži se reža dodeljena A-ju označi kot zasedena
- Ko to zazna A, ve da je vzpostavljena enosmerna povezava z B-jem
- A lahko nato pošlje še zahtevo za obojestransko komunikacijo, nakar b ponovi postopek zahteve po komuniciranju

Kontrola trkov poteka podobno kot pri ostalih protokolih. Če dve postaji zahtevata isto kontrolno režo za komunikacijo, le ta ni dodeljena nobeni postaji in pred ponovno zahtevo obe počakata naključno število časovnih enot. Druga možnost trka nastane, če imata dve postaji že vzpostavljeno povezavo do istega prejemnika in želita obe podatke pošiljati po isti reži. V tem primeru bo prejemnik naključno izbral katere podatke bo prejel, ostali pa bodo morali biti ponovno poslani.

Različne uporabe protokola:

- Za konstanten prenos podatkov postaja lahko prejemniku pove, da bo v vsaki pojavitvi določene reže (te se ciklično ponavljajo) del njenih podatkov.
- Za nepovezavno komunikacijo lahko postaja prejemniku v kontrolnem kanalu le sporoči mesto kjer se nahajajo podatki. Če ima prejemnik to režo prosto, bo podatke prejel, sicer bodo izgubljeni.

Na varianto WDMA protokola obstaja mnogo različic. V njih ima vsaka postaja svoj kontrolni kanal, v drugih je kontrolni kanal le eden. Prav tako je možno protokol realizirati z le po enim nastavljivim sprejemnikom in enim nastavljivim oddajnikom. Poleg naštetih obstaja še veliko drugih različic, ki se razlikujejo v takšnih in drugačnih podrobnostih.

Protokoli za brezžična lokalna omrežja

Tipično brezžično lokalno omrežje (WLAN) ima strateško postavljene bazne postaje (npr. brezžično omrežje Eduroam na FRI) po zgradbi. Bazne postaje so povezane med sabo s paricami ali pa z optičnimi vlakni. Če

omejimo moč signala na 3-4 m, to pomeni, da postane vsaka izmed sob zgradbe svoja celica. Zgradba sama tako postane velik celični sistem, kot pri navadnih celičnih telefonskih sistemih. Tipična pasovna širina je 11 do 54 Mbps.

Sistema **CDMA** ne moremo uporabiti, saj bi v primeru, da je sprejemnik v dosegu dveh aktivnih oddajnikov, prejeti signal bil pomešan in tako neuporaben.

Naivni pristop pri uporabi WLAN bi bil poizkus uporabe **CSMA**, kjer poslušamo, ali kdo oddaja in če tega ne počne noben drug, lahko oddajamo. Težave nam v tem primeru povzročijo motnje pri sprejemniku in ne pri pošiljatelju.

Kot primer lahko pogledamo postaje A,B,C,D. Med sabo sta vidni samo sosednji dve. Tako lahko C, vpliva na B in D.

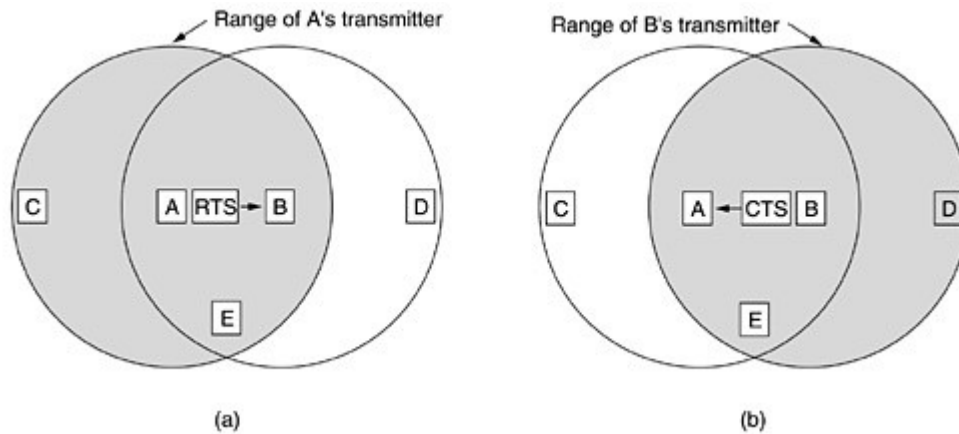


a) Problem skrite postaje: A prenaša k B pri tem pa C tega ne zazna, saj je ta izven dosega postaje A. Če začne C prenos, bo to pokvarilo okvir, ki ga ta dobiva od B. Do problema pride, ker postaja zaradi oddaljenosti ne more zaznati morebitnega tekmeca.

b) Problem izpostavljene postaje: B prenaša k A pri tem pa C prenos zazna. To povzroči, da C napačno predpostavi, da ne sme opraviti prenosa k D. Tak prenos bi vplival le na sprejem med B in C, ki pa nista sprejemnika.

MACA in MACAW

Leta 1990 definiran protokol za WLAN je MACA – *Multiple Access with Collision Avoidance*. Značilnost protokola je v tem, da pošiljatelj opozori prejemnika s kratkim okvirjem **RTS**, da bo začel prenos. Prejemnik se na to odzove s kratkim okvirjem **CTS**. CTS vsebuje dolžino pošiljateljevega sporočila, ki je bila prepisana iz RTS. Prenos se lahko prične ob prejetju CTS okvirja. Izmenjava med pošiljateljem in prejemnikom povzroči, da so sosednje postaje obveščene o prenosu.



–€postaje, ki slišijo RTS, so v bližini pošiljatelja in morajo ostati tiho dokler ta ne prejme CTS

–€postaje, ki slišijo CTS, so v bližini prejemnika in morajo ostati tiho za čas poteka prenosa, katerega dolžino lahko razberejo iz okvirja CTS

Kolizije so možne v primeru, da dobi prejemnik hkrati RTS okvir dveh različnih pošiljateljev. Neuspeli pošiljatelj počaka naključno dolgo časa in na to poskuša ponovno. Algoritem se imenuje „*binary exponential backoff*“.

Leta 1994 je definiran MACAW (*MACA for Wireless*), ki je izboljššan MACA. Dodali so okvir **ACK** za vsakim uspešno prenesenim podatkovnim okvirjem; zaznavanje prenosa; „backoff“ algoritem za vsak podatkovni tok. S temi spremembami je postal protokol pravičnejši. Za izboljšanje učinkovitosti sistema pa so dodali mehanizem za izmenjevanje informacije o preobremenjenosti med bazami in način, da ni odziv „backoff“ algoritma tako buren med občasnimi težavami.

IEEE

Najpomembnejši standardi, ki so danes v uporabi, so:

802.3 (Ethernet)

802.11 (wireless LAN)

802.15 (Bluetooth)

802.16 (wireless MAN)

ETHERNET

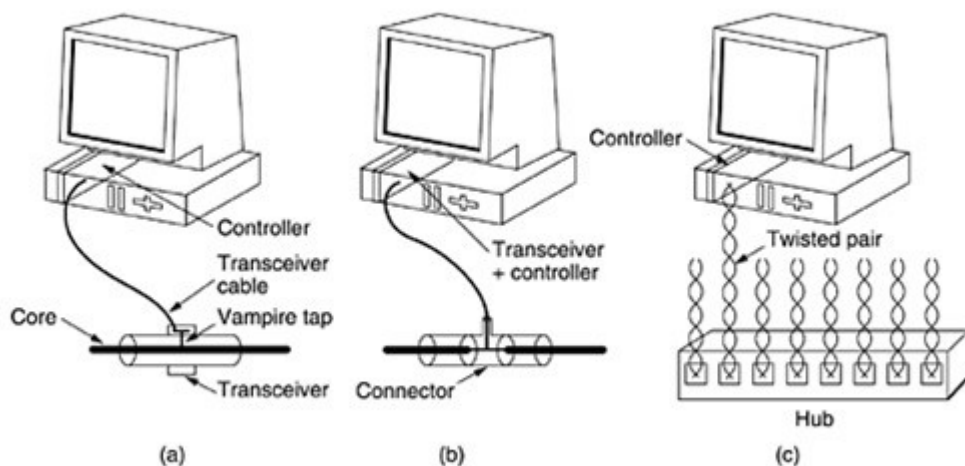
Omrežje Ethernet je standardizirano kot IEEE 802.3. Pojavilo se je leta 1976 (podjetje Xerox) v Kaliforniji, kasneje je bil dograjen. V uporabo je prišel po letu 1980 in je hitro zamenjeval omrežja tipa 'token ring' in 'ARCNET'.

Kot lahko iz spodnje tabele vidimo, je bila prvotna zamisel Etherneta ta, da bi računalniki komunicirali preko skupnega koaksialnega kabla. Takšen kabel so kmalu zamenjali hubi ter stikala, preko katerih povežemo računalnike.

Dolžine segmentov:

Name	Cable	Max. seg.	Nodes/seg.	Advantages
10Base5	Thick coax	500 m	100	Original cable; now obsolete
10Base2	Thin coax	185 m	30	No hub needed
10Base-T	Twisted pair	100 m	1024	Cheapest system
10Base-F	Fiber optics	2000 m	1024	Best between buildings

Prva takšna omrežja so omogočala hitrosti do 10Mb/s, danes pa je ta številka že veliko večja. Najpogosteje se uporablja 100Mb/s ter 1Gb/s, standardiziranih je tudi že 10Gb/s. Omrežje Ethernet lahko zgradimo na osnovi koaksialnih kablov, bakrene parice ali kablov iz optičnih vlaken. Če želimo povečati razdalje, zgornje razdalje za Ethernet, uporabimo ponavljalnike signala.



Pri 10Base5 (število 10 predstavlja hitrost v megabitih na sekundo, število 5 pa razdaljo do 500 metrov) načinu (a) z oddajno sprejemnim kablom (Transceiver cable) povežemo oddajnik-sprejemnik in računalnik. Dolžina takšnega oddajno sprejemnega kabla je lahko največ 50 metrov in vsebuje 5 neodvisnih bakrenih paric.

10Base2 načinu (število 10 predstavlja hitrost v megabitih na sekundo, število 2 pa razdaljo do 200 metrov, a kasneje so ugotovili, da je 200 metrov nekoliko preveč in realno je to 185 metrov) (b) ima tudi svoje prednosti. Če želimo naprimer doma med seboj povezati le 2 ali 3 računalnike, bo cena **nizka**. Vendar paziti je treba, da so vsi kabli pravilno povezani na T-konektorje, hkrati pa je potrebno vse skupaj še ozemljiti. Zaradi teh slabosti pa se ga vse bolj zamenjuje z 10BaseT.

Pri 10Base-T uporabljamo bakreno parico, s katero je potrebno povezati hub ter računalnik. Pomembna prednost tu je, da kabel ni skupen, kot pri prejšnjih načinih. Hkrati tudi ni zapleteno dodajati novih hubov. Poznamo tudi večje hitrosti do 100Mb/s (100Base-T) in več.

Shema pri Ethernetu je enostavnejša kot pri tehnologijah 'token ring' ali 'token bus'. Kadar želi računalnik komunicirati, oziroma pošiljati informacije, uporablja naslednji algoritem:

Glavna procedura

1. Pripravi okvir za prenos.
2. Je medij prost? Če ni, počakaj, da se sprost.
3. Začni prenos.
4. Je prišlo do kolizije? Če je, pojdi na proceduro za detekcijo kolizij.
5. Ponastavi 'retransmission' števec in zaključi prenos.

Procedura za detekcijo kolizij

1. Nadaljuj prenos in poskrbi za to, da bodo vsi prejemniki zaznali kolizijo.
2. Povečaj 'retransmission' števec.
3. Ali je bilo doseženo maksimalno število poskusov prenosa? Če ja, končaj prenos.
4. Izračunaj in počakaj na naključno periodo, ki temelji na številki kolizije.
5. Ponovno vstopi v glavno proceduro na koraku 1.

Pogovarjali smo se o problemu, ki nastopi, ko hoče več naprav naenkrat komunicirati preko enega samega medija. Poznamo modulacije z različnim valovnim signalom vendar pa mora naprava C vedeti kateri del sporočila je rekel A in kaj je rekel B. Zato uporabljamo različne protokole, ki nadzirajo dostop do medija (angl. Medium Access Control).

- MAC je poseben sloj, ki določa kdaj lahko naprava nekaj pošle oz. 'pove'

Dostop do kanala se lahko dodeljuje na dva načina:

Statično:

Delitev kanala na frekvenčne pasove – frekvenčno multipleksiranje, primerno za malo uporabnikov, ki intenzivno komunicirajo. Pri veliko uporabnikih, katero število se spreminja in je promet v rafalih frekvenčno multipleksiranje ni rešitev.

BERNOULIJEVO ZAPOREDJE: v vsakem poskusu iz zaporedja neodvisnih poskusov se mora zgoditi dogodek A z verjetnostjo $P(A)=p$ ali njegova negacija z verjetnostjo $q=(1-p)$. Dogodek A v n zaporednih poskusih se zgodi k-krat na $\binom{n}{k}$ načinov. Verjetnost je $P_n(k)=\binom{n}{k} p^k q^{n-k}$.

Bernoullijeva formula je skoraj neuporabna za računanje velikih n. Zato pri velikih n za njeno aproksimacijo uporabljamo Poissonovo formulo.

Poissonovo formulo pogosto uporabljamo za opisovanje naključnih procesov v strežnih sistemih (angl. queueing systems). Verjetnost, da se ne bo zgodilo nič je $P_0=e^{-\lambda t}$.

Na kratko: Po kanalu sporoča le ena naprava ali pa se lahko izmenjuje več naprav. Vse naprave vedo kdaj kdo govori, saj imajo omejen čas. Težava nastopi v primeru, če nekdo nima nič za 'povedati' in je takrat kanal neizkoriščen.

Dinamično:

- Predpostavke modela:

- Imamo n neodvisnih postaj. Verjetnost generiranja paketa v intervalu Δt je $\lambda \Delta t$, kjer je λ konstanta (povp. št. prihodov novih okvirov).
- Ko je okvir generiran, postaja miruje, dokler ni v celotiposlan.
- Predpostavka obstoja enega samega kanala - če pride do sočasnega oddajanja, nastopi trk.
- Čas: neprenehna možnost oddajanja (zvezen čas) ali le ob določenih intervalih (diskreten t).
- Poslušanje kanala: postaje poslušajo medij, ali pa ne (zaznajo šele po trku, da medij ni prost).

Na kratko: govori tisti, ki ima kaj za sporočiti

Algoritmi za dostop do medija

ALOHA – razvit na univerzi na Havajih leta 1970.

•Delovanje:

Postaje pošljejo podatke, čim so ti na voljo. Po pošiljanju ugotovijo ali je bilo sporočilo uspešno poslano ali pa je prišlo do trka. V primeru trka sporočilo ponovno pošljejo v naključno generiranem času, tako ne pride do neskončnih trkov.

•Poznamo 2 različici: čista in predalčna.

Čista oz. Pure ALOHA: paketi se pošiljajo v povsem naključnih časih. Zato lahko do napak oz. trkov pride na začetku ali na koncu poslanega paketa. Izkoriščenost kanala je nekje 18%.

Predalčna oz. Slotted ALOHA: V tem primeru, celotno časovno os ne gledamo zvezno ampak kot predalčke (slote). Pakete lahko pošljamo samo v svojem predalčku. Izkoriščenost kanala je zato bistveno večja, nekje 38%.

Carrier Sense Multiple Access (CSMA)

- CS (Carrier Sense) - prisluškovanje mediju in odkrivanje zasedenosti komunikacijskega kanala
- MA (Multiple access) - vsak podatek prispe vsem postajam znotraj medija
- Zaznava nosilne frekvence
- Naprave poslušajo medij in oddajajo sporočila le v primeru, če je kanal prost-2 načina:
 - Vztrajna (*princip:* vseskozi poslušaj, sporočaj čim je kanal prost)
 - Nevztrajna različica (*princip:* poslušaj v intervalih, sporočaj čim je kanal prost).

- Obstaja tudi CSMA / CD (Collision Detect):

(princip: prenehanje sporočanja, ko zazna trk)

- Standardizirana različica CSMA/CD protokola je znana pod imenom Ethernet
- Tako dosežemo boljšo izkoriščenost kanala (skoraj 100%)

CSMA ni smiselno uporabiti, če imamo malo 'prometa', saj izgubi na učinkovitosti.

Pojavile so se potrebe po povezovanju posameznih omrežij med sabo zaradi geografske oddaljenosti, optimizacije prometa in večje zanesljivosti ter varnosti. Omrežja lahko med sabo povezujemo z različnimi napravami, ki pa ne delujejo vse na enakem nivoju omrežja. Uporabljamo:

- ponavljalnike,
- hub,
- mostove,
- stikala,
- usmerjevalnike,
- transportne prehode in
- aplikacijske prehode

Ponavljalnik (repeater)

Samo ojača signal, ki pride do njega. To stori s pomočjo Teslove tuljave. Ne razume in ne preverja digitalne vrednosti signala. Deluje na fizični ravni.

Hub

Deluje na fizični ravni. Je nadomestek načina povezovanja, ko so posamezni računalniki povezovali na isti kabel. Ta "kabel" se nahaja v sami napravi, nanj pa so povezani računalniki preko parice. Signale oziroma podatke, ki jih dobi na vhode pošlje na vse izhode in se ne ubada s tem, komu so podatki v resnici namenjeni.

Most (bridge)

Deluje na povezavni plasti. Skrbi za premoščanje prometa iz enega tipa omrežja v drugega. To dela tako, da iz oklvara, ki je prišel iz enega tipa omrežja izlušči dejanske podatke in jih zapakira v paket namenjen ciljnemu omrežju. Npr. z mostom lahko povežemo brezžično in ožičeno omrežje.

Stikalo (switch)

Deluje na povezavni plasti. V sebi skriva tudi ponavljalnik in ker vsebuje še nekaj dodatne logike se ga uporablja več kot ponavljalnike. "Zaveda" se digitalne informacije skrite v signalu. To informacijo preverja,

če je pravilna ipd. Razlika med stikalom in hubom je v tem, da stikalo ne pošilja vhodov na vse izhode, ampak samo na tiste, ki so skupaj v istem omrežju (kot se nastavi v samih nastavitvah stikala).

Usmerjevalnik

Deluje na omrežnem nivoju. Skrbi za ustrezno usmerjanje prometa po omrežju. Izbira najustreznejše poti za posamezne pakete in tako skrbi, da ne bi bila neka pot po omrežju preobremenjena.

Hitri Ethernet (Fast Ethernet)

Najprej se je zdelo, da bo 10mbps dovolj. Kmalu so ugotovili, da to ni res in so nadgradili osnovni Ethernet protokol na 100mbps. Ta način dovoljuje povezave na najnižjem nivoju samo še preko stikal in hub-ov. BNC konektorji in "vampire taps" niso več dovoljeni.

Gigabitni Ethernet (Gigabit Ethernet)

Pojavi se leta 1998. Razvit je bil iz Fast Ethernet in nosi oznako 802.3z. Najprej je bil razvit na optičnem kablu, kasneje pa tudi na bakreni parici vendar na krajše razdalje. Ohranila se je kompatibilnost za nazaj. Novo pa je to, da so vse povezave sedaj tipa točka-točka.

V nadaljevanju se je zgodilo to, da se je povezovalna plast razdelila na dva nivoja, in sicer:

- fizični sloj in
- nadzorni sloj

Zaradi te razdelitve imamo lahko isti protokol na različnih prenosnih medijih. Iz tega izhaja cel kup variant ethernet protokola: 802.3, 802.11,...

802.11 MAC podplastni protokol (802.11 MAC sublayer protocol)

Pri brezžičnih omrežjih je glavni problem da računalniki ne vidijo vseh računalnikov v omrežju, poleg tega pa je večina brezžičnih povezav je half duplex, kar pomeni da nek računalnik ne more istočasno oddajati in poslušati. Zaradi teh dveh problemov ne moremo uporabiti protokola CSMA/CD, ki ga uporablja ethernet. Zato da računalniki vedo ali so pevezave proste se pred začetkom pošiljanja pošlje zahteva za pošiljanje(RTS) in potrdilo da lahko pričnemo s pošiljanjem(CTS), po koncu pošiljanja se pošlje signal za konec pošiljanja(ACK). Nadgradnja tega sistema je, da se velike količine podatkov pošilja po delih.

802.11 struktura okvirja (802.11 frame structure)

Okvir protokola 802.11 je sestavljen iz glave, podatkov in repa.

Glava je sestavljena iz:

- Kontrolnega segmenta,
- dolžine okvirja,
- naslova pošiljatelja,
- naslova prejemnika,
- naslova pošiljateljeve bazne postaje,
- naslova prejemnikove bazne postaje in

- sekvence(v tem segmentu, oštevilčimo okvirje)

Podatkovni del okvirja je dolg 2312 bytov.

V repu okvirja je kontrolna vsota. Na podlagi kontrolne vsote ugotovimo, če se je okvir pravilno prenesel.

Storitve protokola 802.11

Oziroma, kaj je potrebno za delovanje.

- prijava v omrežje,
- odjava iz omrežja,
- povezava z globalnim omrežjem in
- priključitev na globalno omrežje

Za samo delovanje moramo biti na voljo še posamezne storitve v sami celici, da je celica uporabna. Te storitve so:

- prijava z računom (authentication),
- odjava računa (deauthentication),
- privatnost (kriptiranje podatkov) in
- dostava podatkov

Široko pasovno brezžično omrežje (broadband wireless)

Široko brezžično omrežje uporablja standard 802.16. Ta standard so začeli razvijati julija 1999. Dokončan je bil aprila 2002.

Uporablja se za brezžična mestna omrežja(wireless MAN).

Tako kot nekateri ostali 802 standardi je tudi 802.16 bil bod močnim vpivom OSI modela, kar pomeni da ima podobno zgradbo kot ostali 802 standardi.

Razlike med 802.11 in 802.16:

- 802.11 je halfduplex, 802.16 je fulduplex
- ker je doseg 802.16 lahko več kilometrov, ima ta standard več modulacijskih shem
- zaradi velikosti omrežja in števila povezav potrebuje 802.16 večji spekter frekvenc(10 - 66 GHz)

802.16 MAC podplastni protokol (802.16 MAC sublayer protocol)

802.16 struktura okvirja (802.16 frame structure)

Lokalno omreževanje (local internetworking)

Pod tem pojmom se razume, kako je neko lokalno omrežje povezano med sabo s pomočjo večih mostov (za premoščanje lokalnega in povezovalnega omrežja). Za ugotavljanje kje se znotraj posameznih "podomrežij" nahaja nek računalnik oz. kakšna druga naprava, so razvili protokol imenovan ARP. Iz takšnega omrežja je potrebno izločiti ciklje (kaj to je, glej snov APS1) in povezave zgraditi v obliki minimalno vpetega drevesa. To je potrebno storiti zato, da se posamezni okvirji ne pošiljanjo okoli in okoli po omrežju, ampak pridejo na neko končno lokacijo in se tam končajo.

Odaljeni mostovi (remote bridges)

Oddaljeni mostovi se uporabljajo, če želimo povezati med seboj več LAN omrežij, ki so recimo v različnih zgradbah. Mostove povežemo s povezavami točka-točka.

Za povezave med mostovi lahko uporabimo dva različna protokola:

- Prva možnost je, da uporabimo standardni točka-točka protokol kot je PPP, ki vstavi celoten MAC okvir v polje za prenos.
- Druga možnost je, da okvirju odstranimo glavo in rep, in tako pošljemo je podatkovni del. Nova glava in rep pa se izračunata in dodata na ciljnem mostu. Slabost te možnosti je, da ne moremo vedeti, če je prišlo do napake, ker pri ciljnem mostu nimamo kontrolne vsote od začetka prenosa ampak imamo le kontrolno vsoto ki se je izračunala na cilju.

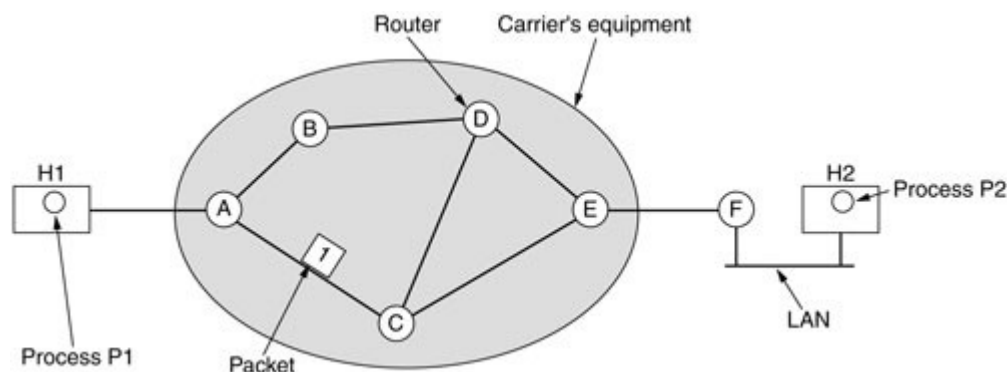
Navidezna lokalna omrežja (Virtual LANs)

Mrežna plast

Mrežna plast:

- omogoča prenos paketkov preko omrežja med dvema postajama, ki nista neposredno povezani med seboj, po optimalni poti

- usmerja pakete med vozlišči v omrežju, zato mora poznati topologijo le-tega



H1 uporabi povezavno plast od H1 do A. Potem se A »odloči«, kako bo spravil podatke od sebe do H2. Uporabi povezavno plast med A in C ali A in B.

- storitve mrežnega sloja so neodvisne od komunikacijskega podsistema
- za transportni sloj so število, tip in topologije neodvisne od LAN in WAN
- omrežni naslovi, ki so na voljo transportnemu sloju, so enotni
- mrežna plast pošilja pakete, ki jih povezavna plast da v okvirje; če je paket prevelik za en okvir ga najprej razdeli na dele velikosti enega okvirja - fragment paketa

Internet:

- podmreža prenaša izključno samo bite
- mrežna storitev je lahko NEZANESLJIVA (kar pošljemo ni nujno da pride do cilja) in NEPOVEZAVNA (ni potrebna vzpostavitev povezave - paketi potujejo po različnih poteh, npr. SEND PACKET/RECEIVE PACKET)
- vsak paket mora vsebovati naslov, saj se prenaša povsem neodvisno od drugih

OSI (X.25):

- mrežna storitev je POVEZAVNA in ZANESLJIVA, zato so prenešeni podatki brez napak in v pravilnem zaporedju
- pred komunikacijo se vzpostavi povezava, ki se potem vzdržuje in na koncu razdre
- pri vzpostavitvi povezave se sloja pogajata o parametrih in kakovosti

Nepovezavno

- vsak paket se na novo usmerja po poti; lahko potujejo po različnih poteh
- takšna omrežja so bolj robustna
- če je več okvirjev in se med pošiljanjem ena točka »sesuje« lahko za neuspešno poslano okvirje izberemo drugo pot
- težje zagotovimo kakovost

Forwarding = Prepošiljanje - točka A dobi paket, in ga pošlje naprej.

Kam pošlje paket je določeno z usmerjevalno tabelo, v kateri prvi stolpec pove cilj paketa, drugi stolpec pa kateri točki naj A pošlje paket (pot).

Povezavno usmerjena postavitev

vzpostavi t.i. navidezni vod (VIRTUAL CIRCUIT – VC)

- lažje je zagotoviti kakovost
- pri vzpostavitvi povezave se sloja pogajata o parametrih in kakovosti

Navidezni vod – pot med napravama mora biti določena preden se začnejo prenašati paketi. Vsaka povezava ima svoj ID. Naslov pošiljatelja in prejemnika ni potreben, saj je dovolj identifikacijska številka povezave. Pri vzpostavitvi povezave se rezervirajo vsi viri.

- Primer s predavanj:

Dva bi se rada pogovarjala med seboj in poslušala glasbo ki jo posluša drug. V tem primeru so:

Parametri: VoIP, Poslušanje mp3.

Kakovost: 64kb/s za VoIP, 192kb/s za mp3.

Razlike med povezavnim in nepovezavnim:

VIDIK	DATAGRAMSKA PODMREŽA	NAVIDEZNA POVEZAVA
vzpostavitev povezave	ni potrebna	zahtevana
naslavljanje	vsak paket ima naslov izvora in ponora	vsak paket ima le kratek ID navideznega voda
informacija o stanju	ni potrebna	potrebna tabela za vsak navidezni vod
usmerjanje	vsak paket neodvisno	določeno ob vzpostavitvi povezave
posledica izpada usmerjevalnika	ni, le začasno izgubljeni paketi	vsil navidezni vodi se razpustijo
zamašitev	težko obvladljiva	enostavno obvladljiva (medpomnilniki)

Učni načrti Računalništva in Informatike ter tekmovanja

"Če se učim uporabljati pralni prašek, še ne pomeni, da se učim kemijo!"

2 agenciji za razvoj:

1. ACM (Association for Computing Machinery)

- Obstaja cca. 50 let
- Je strokovna organizacija, ki skrbi za razvoj stroke računalništva in informatike
- Imajo digitalno knjižnico, kjer je vse znanje, ki je tam (zapiski, poročila, članki,...); FRI ima dostop do le teh
- 2 nivoja:
- revije, poljudnoznanstveni članki,... Zanimivo za študente)
- poročila s konferenc,...
- Podeljujejo Turingovo nagrado (ekivalentna je Nobelovi, samo v računalništvu in informatiki)
- Ima interesne skupine (SIG)
- Računalniška arhitektura, komunikacije
- www.acm.org
- tudi ACM Slovenija (www.acm.si)
- prireja tekmovanja, predavanja,...

2. IEEE (Institute of Electrical and Electronics Engineers)

- Je mednarodna neprofitna profesionalna organizacija za napredek v tehnologiji povezano z elektriko
- Ima največ članov izmed katerekoli tehnične organizacije na svetu (preko 365.000 članov v približno 150 državah)
- Več organizacij – tudi regijske, med katerimi je tudi IEEE Slovenija
- Pod to organizacijo obstajajo tudi društva kot so IEEE Education Society, IEEE Computer Society,...
- Tudi imajo digitalno knjižnico, ki pa je bolj usmerjena v elektrotehniko.

Leta 1991 sta se organizaciji odločili, da se bosta zmenili, kaj *Computer Science* sploh je oz. pomeni. Sta sestrski organizaciji.

Poleg tega se ACM ukvarja tudi z izobraževanjem v šolah (šolski učni programi)

Računalništvo se deli na:

- **Hardware** (Electric Engineering, Computer Engineering)
- **Software** (Computer Engineering, Computer Science, Software Engineering)
- **Organisational**(Information Science, Information System)

Že v osnovnih šolah (OŠ) naj bi začeli z računalništvom in informatiko:

- K-8 (From Kindergarten to 8TH class)
- (9 or 10)TH
- (10 or 11)TH
- (10 or 11)TH – znanje se stopnjuje z vsakim razredom

Tekmovanja:

Univerzitetna(a) in srednješolska(b) raven

a) Svetovna(ACM ICPC), regionalna, nacionalna(UPM), Challenge Cup,...

Usmerjanje

- **Statično**

Vsako vozlišče točno ve kam poslati določen paket za določen naslov. Vse povezave so statično vnaprej določene ob zagonu (lahko jih tudi ročno dodamo). Slabost tega principa je da se ob izpadu neke povezave ta sprememba v omrežju ne upošteva in se paketki še vedno pošiljajo na to povezavo.

- **Dinamično**

Pot po kateri naj paket potuje se vsakič znova izračuna. Računa se s pomočjo podatkov, ki jih posamezni usmerjevalniki dobijo od svojih sosedov ali od vseh ostalih usmerjevalnikov. Ta princip omogoča, da se upoštevajo spremembe v omrežju.

- **Kriteriji za optimalno pot paketkov:**

- o najkrajša pot
- o najcenejša pot
- o najhitrejša pot
- o itd.

- **Princip pravičnosti (fairness) v nasprotju z optimalnostjo**

Optimalnost glede na celoten model lahko privede do tega, da določeni paketi nikoli ne bi prišli do cilja - zato poskrbi princip pravičnosti, ki poskrbi, da se eventualno prenesejo tudi paketi, ki niso na optimalnih poteh.

- **Algoritmi za usmerjanje**

- o Algoritem Najkrajša pot

Tukaj se lahko uporablja Dijkstrov algoritem, kateri s pomočjo požrešne metode izračuna najkrajšo možno pot za vsa vozlišča v omrežju. Ta algoritem lahko najde optimalno pot za nek paketek.

- o Algoritem poplavljanja

Deluje tako, da vsak usmerjevalnik vsak sprejeti paket posreduje na vse svoje sosedo razen na tistega od katerega ga je prejel. S tem pristopom bi eventualno vsa vozlišča dobila poslani paket. Slabost tega algoritma je, da se paketki neskončno podvajajo. Ta problem lahko zmanjšamo s uporabo števca v glavi paketka.

- o Usmerjanje na osnovi vektorja razdalj

Je dinamičen algoritem, pozna tudi kot RIP (Routing Information Protocol). Deluje tako, da vsak usmerjevalnik hrani tabelo (vektor) razdalj do vse ostalih vozlišč. Te tabele vozlišče nato izračuna tako, da vpraša vse svoje sosede kolikšna je njihova razdalja do vseh ostalih vozlišč (razdalje se ugotavljajo s pomočjo Ping orodja) in na podlagi tega zgradi svojo tabelo, najbolj optimalna pot se nato ugotavlja s pomočjo teh tabel. Vsak usmerjevalnik mora svojo tabelo povezav tudi periodično obnavljati. Če neka povezava izgine, traja neskončno mnogo da se to ugotovi, če pa se pojavi nova povezava pa se to ugotovi zelo hitro. Za nadzor in upravljanje omrežja se uporablja protokol ICMP (Internet Control Message Protocol).

- o Usmerjanje na osnovi stanja povezav

Je nekoliko boljši način usmerjanja od usmerjanja na osnovi vektorja razdalj, saj lahko v bližnjem omrežju dokaj hitro ugotovi, če je kaka povezava padla. Stanje posameznih povezav ugotavlja s pomočjo »učenja od sosedov«.

- o Hierarhično usmerjanje (BGP in IGP)

Internet je razdeljen na posamezne otoke. Usmerjevalniki so razdeljeni na tiste ki skrbijo za povezavo med temi otoki in na tiste kateri skrbijo za povezave znotraj otoka.

- o Usmerjanje z oddajanjem (Broadcasting)

Posamezne paketke pošljemo vsem sosednjim vozliščem. Sosedje paketke pošljejo svojim sosedom, razen tistemu od katerega so dobili paketek. Problem pri tem načinu usmerjanja je preveliko podvajanje paketkov. To lahko rešimo s pomočjo hranjenja zgodovine določenega števila prejšnjih paketkov v vsakem usmerjevalniku. Ko pošljemo več zaporednih paketkov na isti naslov se lahko tako pot določi iz zgodovine prejšnjega paketka. Druga rešitev je števec v glavi paketa, ki pa se ob vsakem prenosu zmanjša. Ko ima števec vrednost 0, se prenos prekine.

- o Večsmerno usmerjanje (Multicasting)

Paketek pošljemo samo vsem tistim vozliščem, ki so se naročili na podatke. V glavo paketka se tako doda naslove tistih vozlišč, katerim se naj pošlje. Vsak usmerjevalnik potrebuje seznam, ki hrani katera vozlišča so naročena na podatke. Za ta način uporabljamo protokol IGMP (Internet Group Management Protocol).

- o Usmerjanje za mobilne komunikacije

To usmerjanje se uporablja, ko imamo premične goste (npr. mobiteli). Ko pridemo v neko tuje omrežje, najprej sporočimo temu omrežju svojo identiteto in katero je naše domače omrežje. To tuje omrežje, nato potrdi našo identiteto z našim domačim omrežjem, dodeli nam nek trenutni ip, in sporoči domačemu omrežju da se sedaj nahajamo pri njih in da imamo ta trenutni ip.

- o Usmerjanje v Ad Hoc omrežju

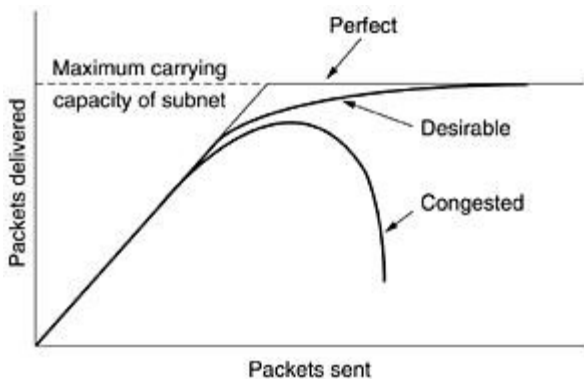
Primer takega omrežja je, ko se tanki vozijo po terenu in komunicirajo med seboj. Faza odkrivanja poteka tako da, vsak tank pošlje broadcast vsem tankom okoli sebe ki so dovolj blizu. Tej tanki nato ponovno pošljejo to sporočilo naprej vsem ostalim, ki so dovolj blizu. Na podlagi teh sporočil se nato gradijo usmerjevalne tabele.

- **Skalabilnost omrežja**

Če povečamo količino prometa, se mora sistem še vedno dobro odzivati. S povečanjem prometa pa zelo hitro napolnimo zgodovino (pri rešitvi z hranjenjem zgodovine), zato bi lahko uporabili rešitev s števcem, ki je skalabilna. Tukaj pa nastane problem, saj če se promet poveča, ne vemo kolikšna mora biti začetna vrednost števca. Rešitev tega problema je, da uporabljamo obe rešitvi skupaj.

Nadzor zamašitev / congestion control

Zamašitev se pojavi, kadar se v neki točki omrežja nabere toliko prometa, da ta skozi to točko ne more več normalno teči. Nadzor zamašitev se od krmiljenja pretoka razlikuje po tem, da velja za celo podomrežje, krmiljenje pretoka pa le med dvema točkama.



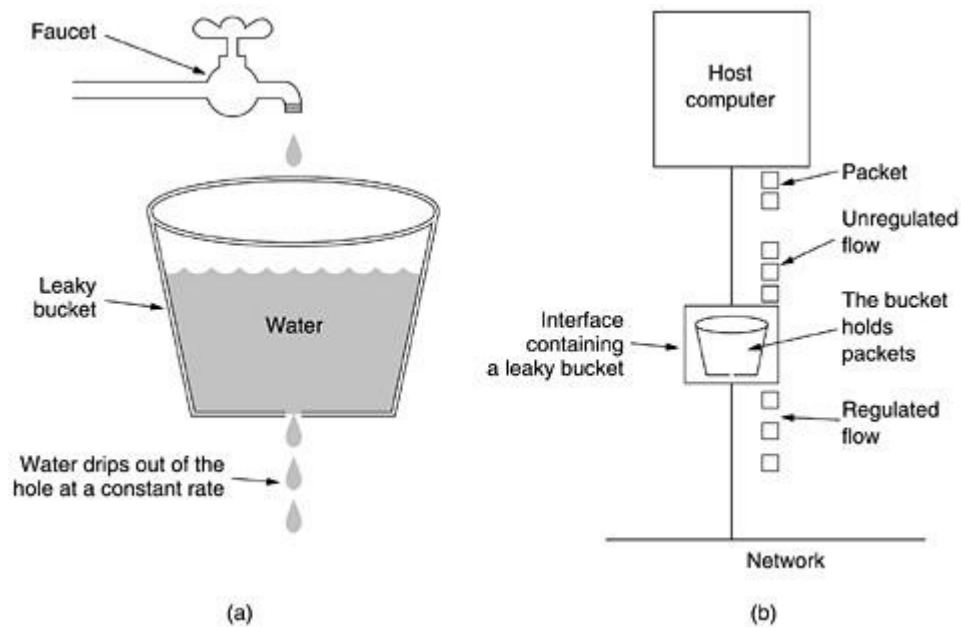
Razlogi za pojav zamašitev:

- množica paketov z več linij se v neki točki združi v eno linijo
- predolgo hranjenje paketov v medpomnilniku in ponovno pošiljanje kot posledica
- prepočasni usmerjevalniki
- premajhna pasovna širina

Glavna principa nadzora zamašitev:

- princip odprte zanke (statično) – v bistvu pomeni dobro načrtovanje sistema - v naprej predvidimo obremenjenost
- princip zaprte zanke (dinamično) – sistem skrbno opazujemo, ko opazimo zamašitev, pošljemo informacije o tem ostalim točkam in spremenimo način delovanja omrežja.

Nekaj pristopov k nadzoru zamašitev:



- pristop z glajenjem / buffering (preluknjano vedro / leaky bucket algoritihm)
- z virtualnimi podomrežji – točke, ki so zamašene začasno izločimo iz sistema in pakete usmerjamo po drugi poti
- zamašena postaja pošlje pošiljateljem (ali pa le sosedom) „choke“ paket, ki pove, da je zamašena in naj se paketi usmerjajo po drugi poti

Proti zamašitvam se borimo na več plasteh:

Layer	Policies
Transport	<ul style="list-style-type: none"> • Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy • Timeout determination
Network	<ul style="list-style-type: none"> • Virtual circuits versus datagram inside the subnet • Packet queueing and service policy • Packet discard policy • Routing algorithm • Packet lifetime management
Data link	<ul style="list-style-type: none"> • Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy

IP protokol je dobro razširljiv zato, ker postajam pri prenosu ni potrebno vedeti veliko o podatkih, ki potujejo skozi. Hkrati pa tu leži del problemov pri reševanju zamašitev.

Kakovost storitev / Quality of Service

Odvisno od storitve:

Application	Reliability	Delay	Jitter	Bandwidth
E-mail	High	Low	Low	Low
File transfer	High	Low	Low	Medium
Web access	High	Medium	Low	Medium
Remote login	High	Medium	Medium	Low
Audio on demand	Low	Low	High	Medium
Video on demand	Low	Low	High	High
Telephony	Low	High	High	Low
Videoconferencing	Low	High	High	High

Trepanje / jitter (velike razlike v časih prenosa) in zamiku / delay rešujemo s glajenjem / buffering.

Packet Scheduling

Imamo usmerjevalnik z večimi vhodi in enim izhodom (glej tudi: prvi razlog za zamašitev)

- če bi radi, da so vsi vhodi enako zastopani uporabimo pravično postavljanje v vrsto / fair queueing
- lahko definiramo multicast skupine – ena postaja oddaja, več pa jih prejema, s tem, da podatkov ne podvajamo v omrežju (IGMP),
- lahko vzpostavimo navidezne povezave, kjer rezerviramo del zmogljivosti neke poti po omrežju, a ta rešitev ni skalabilna. (RSVP – **ReSerVation?** Protocol, del združenih storitev / **IntServ?**).

- Imamo tudi razdružene storitve / **DiffServ?**, kjer imamo osem nivojev, v resnici pa zares potrebujemo le dva
 - o Navadni / Regular, Best-effort
 - o Pospešeni / Expedited
- Pri **DiffServ?** poznamo tudi zagotovljeno / assured usmerjanje, ki definira štiri nivoje.

Katere pakete zavržemo, ko se medpomnilnik usmerjevalnika napolni?

- Vinska odločitev
 - o obdržimo stare pakete, ki morda za naslovnika nimajo več smisla
- Mlečni pristop
 - o shranjujemo novejše pakete, stare pa mečemo stran
- Vmesna varianta
 - o novejše pakete mečemo stran naključno (s tem lahko stran vržemo tudi starejšega)

Label Switching MPLS

V paketu imamo oznake, ki ohranjanje neke vrste navidezno povezavo do naslovnika. Ta rešitev je bolj skalabilna od RSVP, kjer te podatke hranijo postaje na poti.

Slovar

- PETS - TPDU
- algoritem preluknjano vedro - the leaky bucket algorithm
- nadzor zamašitve - congestion control
- krmiljenje pretoka - flow control
- hrbtenično omrežje - backbone
- izvor - source
- ponor - destination
- plast - layer
- sloj - layer
- vtičnica - socket
- vrata - port
- pasovna širina - bandwidth
- podatkovno breme - data payload
- zakasnitev - delay
- krožni čas potovanja - round-trip time (RTT) *Se priporočam za boljši prevod (obhodni čas?)*
- zakasnitev krožnega potovanja po omrežju - round-trip delay
- življenjska doba paketa - packet lifetime
- produkt zakasnitve in pasovne širine - bandwidth-delay product
- dvojno rokovanje - two-way handshake

- trojno rokovanje - three-way handshake
- potrditev - acknowledgment
- smrtni objem - deadlock
- zlog - byte (8 bitov)
- bajt - byte (8 bitov)
- oktet - octet (8 bitov)
- enosmerni - simplex
- izmenično dvosmerni - half-duplex
- popolnoma dvosmerni - full-duplex
- zvita parica - twisted pair
- konstelacijski vzorec - constellation diagram
- hitrost vzorčenja - sampling speed
- širjenje signala po fizičnem mediju - propagation
- PSP - ponovno selektivno prenašanje
- pipelining - cevovodenje
- MAC = Media Access Control
- ARP = Address Resolution Protocol
- komunikacijska številka - connection id

I. domača naloga

1. naloga

V našem prehodu skozi računalniške komunikacije in omrežja bomo spoznali tudi Petra Zmedo. Peter je bivši študent, ki pa je predmet RKO opravljal bolj z levo roko in tako nemalokrat zaide v težave pri svojem delu.

1. Tokrat se je po telefonu pogovarjal s prijateljem Miho. Miha ga je prosil, če mu Peter posreduje podatke, ki jih ima na CD-jih. Podatki zasedajo kar 512 CD-jev in na vsakem CD-ju je 700 MB podatkov. Peter je sedaj pred dilemo, ali naj podatke pošlje po omrežju ali naj preprosto odnese vse skupaj Mihi. Kaj svetujete Petru? Utemeljite svoj odgovor. (*Namig*: ker niso podani niso vsi podatki potrebni za izračun, si manjkajoče podatke smiselno dopolnite).
2. Kakšna je razlika med povezavno in nepovezavno komunikacijo? Naštejte **dve** dobri lastnosti prvega načina komuniciranja in **dve** dobri lastnosti drugega načina.
3. Standardi v Internetu so določeni s pomočjo RFC-jev. Kaj točno pomeni RFC? Kateri RFC določa opisovalni jezik HTML? Kaj je IETF in kakšna je njegova vloga?

2. naloga

1. Na predavanjih smo govorili o Nyquistovi frekvenci oziroma o njegovem izreku. Menite, da velja izrek tudi pri medplanetarni komunikaciji? NASA je poslala na planet Mars dve premična robota, s katerima je tudi komunicirala. Recimo, da bi bila vaša naloga, da predvidite ukaze, ki jih bi bilo možno pošiljati robotu na Mars. Kakšne ukaze bi predlagali - na primer posamične ukaze kot so *obrni se desno, obrni se levo* ali kaj drugega? Utemeljite odgovor. (*Namig*: pomislite, koliko časa potuje informacija na Mars in nazaj).
 2. Peter Zmeda je na predavanjih spoznal konstalacijske diagrame in prišel je do enkratne ideje, da bi jih lahko uporabil tudi pri optičnih povezavah. Je to smiselno? Recimo, da imamo po modemski liniji prenost 56kbaud in QAM-16 konstalacijski diagram za modulacijo. Koliko bitov lahko prenesemo na sekundo?
 3. Peter je tudi zaposlen v podjetju Butale kabel (BK). BK ponuja svojim naročnikom kabelsko televizijo in v zadnjem času tudi Internet. Peter je slišal, da se na tržišču ponujajo trojčki (TV, telefonija in internet). Telekomunikacijske družbe preprosto nadgradijo svoje PSTN tako, da preko njih povezujejo stranke v internet ter nato preko interneta ponujajo vse tri storitve. Tudi Peter bi rad naredil nekaj takšnega. Kako naj se tehnično loti ponujanja vseh treh storitev? Recimo, da je na najdaljši veji zaporedno priklapljenih 64 strank, kolikošno pasovno širino lahko največ ponudi vsaki stranki in koliko v povprečju?
-

3. naloga

Na predavanjih in na vajah smo govorili o arhitekturi oz. topologiji Interneta. Omenili smo tudi, da so lahko iste lokacije povezane z različnimi povezavami in da podatkovni paketi ne uberejo vedno iste poti.

1. Poiščite kakšen zemljevid prekooceanskih povezav in si oglejte kje potekajo povezave med celinami.
 2. Katera organizacija skrbi za dodeljevanje naslovov IP v Evropi? S katero storitvijo si lahko pomagate pri določanju države iz katere izvira nek naslov IP?
 3. Poskušajte posikati čim več povezav med evropsko in afriško celino. Za iskanje povezav uporabite program Traceroute (tracert v okolju Windows). (*Namig*: poiščite univerze na različnih delih afriške celine in poskušajte odkriti pot do njih. Upoštevajte, da so povezave včasih satelitske.)
-

4. naloga

Rdeča kapica je ugotovila, da njena banka Volkbank ne nudi storitev elektronskega bančništva. Ker preveč časa porabi že treking po gozdu in nabiranje zaščitenega cvetja (in hranjenju gozdnih živali, čeprav je to strogo prepovedano), je direktorju Volkbanke, g. Volku, zagrozila s selitvijo vseh svojih denarnih sredstev na Kajmanske otoke. G. Volk se je ob grožnji tako pomembne stranke nemudoma odločil, da Volkbank potrebuje elektronsko bančništvo.

Pomagajte g. Volku pripraviti predloge sporočil XML, ki bodo omogočale izmenjavo podatkov med komitenti in banko. Rdeča kapica je v grožnji omenila, da želi elektronsko uporabljati najmanj naslednje storitve:

1. dvig denarja z računa,
2. polog denarja na račun,
3. odpiranje novega računa,
4. preverjanje stanja na računu.

Za vsako od storitev razmislite katere podatke si morata banka in komitent izmenjati. Upoštevajte, da ima vsako sporočilo tudi ustrezen odgovor (npr. zahteva za preverjanje stanja vrne odgovor, ki vsebuje trenutno stanje na računu). Razmislite, katere podatke boste predstavili kot elemente XML in katere podatke kot attribute elementov XML. Pazite tudi na primere, ko transakcija ni možna (recimo dvig denarne vsote, ki presega limit računa).

Napišite program v Javi, ki bo iz datoteke XML prebral zahtevo in na ekran izpisal vnaprej pripravljen (smiseln) odgovor na to zahtevo v obliki XML. Pri delu z XML:

- uporabljajte le SAX in DOM parser, ki sta vgrajena v JDK, ne uporabljajte drugih knjižnic (WS, SOAP, avtomatsko delo s JavaBean-i, ..),
- ne gradite XMLjev "na roko", ampak uporabite ustrezne metode.

Pri programiranju se držite:

- pravil dobrega kodiranja v Javi (Java coding standard, ...),
- program naj ne vsebuje mrtve kode (skelete primerov ustrezno dopolnite!).

II. domača naloga

1. naloga

1. Črkovni niz *RKO* zapišite v ISO-8859-2 kodu in v Unicode kodu.
2. Niz v obeh kodih zapišite v bitni obliki in s Hammingovimi zaščitnimi biti.
3. Niz v obeh kodih zapišite v bitni obliki in s CRC biti, kjer naj bo polinom x^3+1 .
4. Recimo, da je naš prenosni kanal tako pokvarjen, da vsak četrti bit vedno postavi na vrednost 0 (če je bil pred tem že nič, se nič ne zgodi, če pa je bil 1, postane 0). Kaj se zgodi z našima bitnima nizoma iz 2. in 3. vprašanja? Ali zaznamo napako? Ali jo lahko odpravimo? Svoje odgovore utemeljite z izračunom.
5. Podana je [izvorna koda](#), ki implementira protokol 5 - pojdi nazaj za n (*go back n*). Ali vplivamo na pravilnost delovanja in na učinkovitost protokola, če v funkciji *between* namesto pogoja $a \leq b < c$ uporabimo pogoj $a \leq b \leq c$? Svoj odgovor obrazložite. *Namig: Funkcija between je uporabljena na krožnem vmesniku, o katerem smo se pogovarjali na predvanjih.*
6. Podana je [izvorna koda](#), ki implementira protokol 6 - izbirno ponavljanje (*selective repeat*). Protokol pri sprejemu okvirja preveri, če se zaporedna številka razlikuje od pričakovane in, če je spremenljivka *no_nak* nastavljena na *true*. Če sta oba pogoja izpolnjena, pošlje NAK, v nasprotnem primeru pa zažene časovnik (*timer*). Kako se spremeni pravilnost protokola, če odstranimo stavek *else* (tistega, ki sproži časovnik)? Svoj odgovor obrazložite.

2. naloga

1. V čem je razlika med navadnim in predalčnim Aloha sistemom? Kaj je rezultat te razlike? Kdaj bi uporabili čisti Aloha sistem?
 2. Recimo, da imamo običajno Ethernet povezavo (802.3) in po njej v enem okvirju pošljemo niz *RKO* zakodiran v ISO-8859-2 kodu. Kaj točno potuje po povezavi? Svoj odgovor opišite in narišite. Bodite kar se dá natančni.
 3. Imamo [VLAN omrežje](#) s tremi mostovi (*BR1*, *BR2* in *BR3*) in različnimi (Ethernet) mrežami: 802.3, 802.11, 802.16. Med različnimi vozlišči potuje samo niz *RKO* v podatkovnem delu okvirja. Za vsako od naslednjih potovanj: i) opišite kje potuje; in ii) kako izgleda oziroma se spreminja okvir, ko potuje po VLAN mreži.
 1. A -> B
 2. C -> D
 3. E -> D
 4. A -> F
-

3. naloga

Rdeča kapica (A), babica (B) in volk (C) so vzpostavili lokalno omrežje (za potrebe te naloge, so prijatelji, saj se je volk najedel že v sosednji vasi). Računalnike so najprej povezali z razdelilnikom (hub), nato pa jih je zanimalo še kako se omrežje obnaša, če je povezano s stikalom (switch). Promet med računalniki so spremljali s programom [Wireshark](#), ki ga lahko za branje prometnih datotek uporabite tudi sami.

1. V datoteki [promet - hub B](#) lahko najdemo dve vrsti sporočil ping in odgovorov nanje. Prva so sporočila namenjena enem samemu prjemniku, druga pa so namenjena celotnemu omrežju (broadcast). Poiščite zadnji paket prve vrste in prvi paket druge vrste. V čem se paketa razlikujeta?
 2. Koliko različnih pogovorov (conversations) je prikazanih v seznamu pogovorov (Statistics/Conversation) pri datoteki [promet - hub B](#). In koliko pri datoteki [promet - hub C](#)? Zakaj prihaja do razlik?
 3. Primerjajte datoteki [promet - hub B](#) in [promet - stikalo B](#). Poiščite pakete, ki jih babica (B) vidi, če je omrežje povezano z razdelilnikom (hub), če je povezano s stikalom (switch) pa ne vidi. Razložite zakaj se ti paketi pri povezavi s stikalom ne pojavijo.
-

4. naloga

Babica je že v letih, volk pa je odlomastil nazaj v gozd, le Rdeča kapica še vztraja in bi se rada podučila še kaj več o povezavni plasti. Pri razmišljanju je naletela na nekaj vprašanj, pri katerih potrebujete vašo pomoč.

1. V datoteki [format okvirja](#) si lahko ogledate okvir Ethernet II in okvir IEEE 802.3. Katero polje nam omogoča razlikovanje med protokoloma? Kakšni sta vrednosti te oznake?

2. Oglejte si še datoteko [brežžično brez WEP](#). Kakšni so okvirji povezavne plasti v brezžičnem omrežju IEEE 802.11? Kako se razlikuje od "ožičenega" IEEE 802.3 okvirja?
3. Za operacijski sistem, ki ga uporabljate, poiščite ukaz, ki vam izpiše tabelo ARP. Zapišite in komentirajte njeno vsebino. Kdaj se v tabelo ARP doda nov zapis? Kdaj se zapisi iz tabele brišejo?

III. domača naloga

1. naloga

Usmerjanje in razpošiljanje. Imamo naslednje omrežje:

Recimo, da bi vozlišče F sedaj rado poslalo paket do vozlišča I ter zato mora naračunati pot do I.

1. Recimo, da to izračuna s pomočjo razpošiljanja. Tako razpošlje najprej paket s TTL 1, nato 2, 3 in tako naprej. Kaj se zgodi v vsakem krogu razpošiljanja - opišite, kaj se dogaja v vsakem krogu in kako izgleda usmerjevalna tabela pri F po vsakem krogu?
2. Recimo, da imata vozlišči D in E naslednji usmerjevalni tabeli:

do	D	E
A	5	10
B	2	13
C	1	14
D	0	15
E	15	0
F	-	-
G	7	8
H	8	7
I	6	11
J	9	16
K	6	18
L	4	13
M	2	17

Čemu sta vnosa pri vozlišču F v obeh tabelah - (nedefinirano)? Sedaj F zahteva usmerjevalne tabele pri vseh neposrednih sosedih in ju dobi ter izračuna svojo usmerjevalno tabelo. Recimo, da pred tem F izmeri razdalji do D in E ter dobi vrednosti 3 in 2. Kako izmeri F razdalji do D in E (s pomočjo katerega protokola - opišite postopek in izgled posameznih paketov)? Kako izgleda končna usmerjevalna pri F?

Recimo, da vozlišče B razpošilja promet - npr. B je radijska ali televizijska postaja.

1. Izračunajte minimalno vpeto drevo s korenem v B in so vse vrednosti kakovosti povezav enake (prikažite svoj izračun).
2. Kolikšna je povprečna potrebna pasovna širina pri razpošiljanju radijskega programa in kolikšna pri razpošiljanju TV programa (pri odgovoru navedite vir)?

Recimo, da trenutno poslušajo/gledajo program na vozliščih B, K in M.

1. Kolikšna je povprečna potrebna pasovna širina na povezavi A-B, če se upošteva prej naračunano minimalno vpeto drevo in:
 1. za radijski promet, če B pošilja pakete neposredno vsakemu naročniku;

2. za TV promet, če B pošilja pakete neposredno vsakemu naročniku;
 3. za radijski promet, če B uporablja multikast razpošiljanje;
 4. za TV promet, če B uporablja multikast razpošiljanje.
2. Recimo, da B uporablja multikast razpošiljanje ter bi se na radijski program rad naročil tudi J. Kako to izvede (opišite uporabljeni protokol in posamezne pakete za prijavo ter kaj se bo zgodilo)?
 3. Poleg multikast protokola lahko uporabimo tudi P2P način posredovanja sporočil. Kako deluje? *Namig:* tako deluje izdelek *Octoshape*.
-

2. naloga

Upravljanje prometa in kakovost storitev (QoS). Pri delovanju omrežja uporabljamo princip/algoritem RED. Čemu ga uporabljamo in kako deluje?

Recimo, da v zgornjem omrežju A pošilja tok podatkov v L in pri tem uporablja princip premičnega okna (*sliding window*) - gl. povezavno plast. Pri tem uporablja še naslednji princip:

- če dobi potrditev prenešenega prometa, podvoji velikost okna (do neke največje vrednosti); in
- če ugotovi, da paket ni bil prenešen, prepolovi velikost okna

Usmerjevalnika D in I uporabljata RED princip ter eden od njiju zavrže paket poslan od A proti L. Kaj se zgodi? Opišite čim bolj natančno - če bo šlo, podkrepite svojo razlago z matematičnimi izrazi.

Recimo, da je povezavi A-B in A-G ethernet povezava IEEE 802.3 ter, da A pošilja L pakete velike 2KB. Kaj se zgodi? Opišite postopek in obliko posameznih paketov. Ali morda zaradi tega L opraviti kakšno dodatno delo?

Recimo, da sedaj A uporablja za pošiljanje diffserv storitev. Kje v IP paketu se to odraža (posebej prikažite za IPv4 in za IPv6)? Ali menite, da to kaj vpliva na delovanje vozlišč D in I ter RED principa v njima?

3. naloga

Ker je Rdeča kapica skupaj z družino, prijatelji in sovražniki na dopustu, smo vam za tokratno domačo nalogo, tako kot vsako leto, v sodelovanju s podjetjem [Nil](#) omogočili reševanje vaje *IP primer*. Gre za vajo, ki pokriva teoretična znanja iz IPv4 in laboratorijske naloge iz področja konfiguracije omrežne opreme [Cisco](#).

Za reševanje naloge se prijavite na spletni strani podjetja [Nil](#). Pri registraciji izberite možnost **I'm a student (FRI, FE, FERI, PeF)** in v vnosno polje **Reg. number** vnesite svojo vpisno številko. Vaše sezname smo že posredovali na Nil, tako da pri prijavi ne bi smeli imeti težav. Če sistem vaše vpisne številke ne najde, izberite možnost **Other** in vnesite svoje podatke (vnesti morate vsaj vaše ime in priimek, da bomo našli vašo oceno).

Vaja IP Primer je sestavljena iz teoretičnega in laboratorijskega dela. Teoretični del obsega 3 module (za dostop do povezav morate biti prijavljeni v Nil-ovo domačo stran):

1. [Internetworking concepts and internet protocol](#)
2. [Internet protocol features and functions](#)
3. [Configuring Cisco routers for ip](#)

Glede na to, da snov, ki jo pokrivata modul 1 in 2 že obdelamo na predavanjih in vajah, preučite le 3. modul, ki vam bo v pomoč pri reševanju laboratorijskih nalog (seveda pa ni narobe, če znanje osvežite še z moduloma 1 in 2). Na koncu vsakega od modulov je preverjanje znanja (Assesement test). Testa vam ni potrebno reševati (ne bomo pa vam šteli za slabo, če se lotite reševanja).

Reševanje laboratorijskih domačih nalog poteka oddaljeno na omrežni opremi v podjetju [Nil](#). Do opreme pridete tako, da s pomočjo žetonov, ki so vam na voljo (na voljo imate 3 žetone) na povezavi **Remote lab** najprej rezervirate opremo. Izberite si termin v katerem je oprema prosta in ki vam ustreza (vnaprej vas opozarjamo, da ne odlagajte reševanja naloge na zadnji dan, saj se lahko zgodi, da bo vsa oprema zasedena). V izbranem terminu lahko začnete z reševanjem laboratorijskih nalog.

Rešite **vajo LAN**.

4. naloga

V okviru vaje *IP primer* rešite še **vajo WAN**.